



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/723,755	11/26/2003	Keith B. Stobie	13768.459	9956
47973 7590 07/22/2008 WORKMAN NYDEGGER/MICROSOFT 1000 EAGLE GATE TOWER 60 EAST SOUTH TEMPLE SALT LAKE CITY, UT 84111				
			EXAMINER WEI, ZHENG	
			ART UNIT 2192	PAPER NUMBER
			MAIL DATE 07/22/2008	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary**Application No.**

10/723,755

Applicant(s)

STOBIE ET AL.

Examiner

ZHENG WEI

Art Unit

2192

Period for Reply -- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 17 April 2008.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 2-20, 22-40 and 42-44 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 2-20, 22-40 and 42-44 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 23 November 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
- Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
- Paper No(s)/Mail Date _____
- 5) ☐ Notice of Informal Patent Application
- 6) ☐ Other: _____

DETAILED ACTION

Remarks

1. This office action is in response to the amendment filed on 04/17/2008.
2. Claims 15-17 and 42-44 have been amended.
3. The objection to claims 15, 16 and 44 is withdrawn in view of Applicant's amendment.
4. Claims 2-20, 22-40 and 42-44 remain pending and have been examined.

Response to Arguments

5. Applicant's arguments filed on 04/17/2008, in particular on pages 13-16, have been fully considered. For example:

- At page 14, third paragraph, the Applicants argue that the APA teaching "stress test simply ignores any testing output if [the] system doesn't crash when the insert record object is run" implies that output was produced and recorded. Moreover the use of the term "produced" implies recorded or the production of "recorded output".

However, the Examiner respectfully disagrees. First of all, it should be noted that there are two software programs running as disclosed in current application: software under test (database application) and testing program (stress testing/first/second verification level test case). The produced or recorded output as the Applicants argued is generated by the software under

test (database application), not by the testing program (test case). Moreover, as the APA disclosed "... stress test case simply ignores the output produced or recorded [by the database] when running the test case, i.e., the system does not analyze the output" [emphasis added] (see for example, paragraph [0009]).

Therefore, it implies that the output is not recorded/saved as the system does not analyze the output. Accordingly, APA does disclose the limitation that stress test does not produce recorded output as the Applicants argued.

- At page 14, the fourth and fifth paragraphs, the applicants submit that the recited limitation about "selection of the second verification level causes the one or more test cases to invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and present" fails to identify the prior art teaching. The Examiner agrees and the rejection to said claims is withdrawn.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 13-17, 25, 39 and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (Johnson et al., US 2004/0073890 A1) in view of Prabhakaran (US 6,859,758) in further view of Bourne (Kelley C. Bourne, Testing Client/Server Systems) and the admitted prior art (APA) of paragraph [0007] of Applicant's background.

Claim 17:

Johnson discloses, in a computer system that includes software under test, a method of verifying the software with one or more tunable test cases that are capable of being set to any of a plurality of verification levels, the method comprising steps for:

- loading one or more test cases that include a plurality of software testing instructions organized as a plurality of verification levels within a verification hierarchy, wherein at least two verification levels within the verification hierarchy define different amounts of testing to perform for determining if the software functions as intended when executed (see for example, Figure 2, from step 32, "Test Engineering" to step 34, "Project Engineering", "Test Cases" and related text);
- receiving verification setting instructions for one or more desired verification levels from within the verification hierarchy for use in testing the software, wherein the received verification setting instructions select the one or more desired verification levels from a group of verification levels that include at least first and second verification levels, (see for example, Figure 2, step

about passing "Configuration Information" to step 34, "Project Engineering" and related text); and

- testing the software at the one or more desired verification levels, which include at least one of the first and second verification levels, by running the one or more test cases that include the plurality of software testing instructions that correspond to the one or more desired verification levels (see for example, Figure 2, step 36 "Project Testing" and related detailed steps and text).

But does not explicitly disclose wherein selection of the first verification level causes the one or more test cases to be run during testing and which includes invoking an insert record object to determine if the invocation of the insert record object results in a system crash and while refraining from producing any recorded output, and wherein selection of the second verification level, which is distinguished from the first verification level causes the one or more test cases to be run with different instructions and invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and presented when the one or more test cases corresponding to the second verification level are run and such that the recorded output which is produced in response to the one or more test cases being run following the selection of the second verification level is refrained from being produced in response to the one or more test cases being run following the selection of the first verification level.

However, Prabhakaran in the same analogous art of software testing discloses a method and system for stress testing database storage wherein selection of the stress test causes the one or more test cases to invoke an insert record object and to additionally verify through recorded output that a record corresponding to the insert record object was properly inserted and presented (see for example, col.7, lines 36-44, "data be stored in a database", "notifies stressing software 440 that the information has been successfully stored in the database"; also see Fig.4, item 450 "Monitor software" and related text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to include Prabhakaran's test case into Johnson's test management system. One would have been motivated to do so to simplify test case and configuration re-use as suggested by Johnson (see for example, ABSTRACT, "simply test case and configuration re-use").

Prabhakaran further discloses selection of the test case causes the one or more test cases to be run during testing and which includes invoking an insert record object to determine if the invocation of the insert record object results in a system crash (see for example, col.7, lines 45-58, "maximum rate of operations to enterprise storage system 410 is achieved").

But Johnson and Prabhakaran do not explicitly disclose the limitation about refraining from producing any recorded output.

However, APA discloses a concept of stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for

example, paragraph [0009]) and Bourne also discloses detailed information about "stress test" is that "stress testing determines if the system will break down or otherwise malfunction when it is being overloaded" (see for example, p.356, section 11.1 Stress Testing, second paragraph) Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Prabhakaran and Johnson's test case for only focusing on the condition of system crash without producing any recorded output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], "the system does not analyze the output"). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph [0009], "Producing the output in the first place, however, impacts the system, so it would be better not to produce it in the first place.") and also suggested by the Bourne's purpose of the stress testing to determine system crash as addressed above.

Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to combine Prabhakaran, APA and Bourne's teachings into Johnson's system and using Johnson's test case management and customization feature to customize Prabhakaran's stress test case to test database storage using first/second verification level as addressed above according to different requirement functional test (successfully stored in

Art Unit: 2192

database) or system crash test (maximum rate of operations) as disclosed above.

Claim 13:

Johnson further discloses the method of claim 17, wherein at least a portion of at least one of the plurality of software instructions determines that software information is available and uses the information for troubleshooting the software if it is determined that the software does not function as intended when executed (see for example, Figure 2, step 3-5 of "Project Testing 36", "Record Results", "Report Issues", "Provide Test Case Feedback when necessary" and related text).

Claim 14:

Johnson also discloses the method of claim 13, wherein the software information available is debug information (see for example, Figure 2, step 3-5 of "Project Testing 36", "Provide Test Case Feedback when necessary" and related text, also see, p.3, paragraph [0023], "As tests are run and results recorded, report are issued to test engineering for tracking test progress and adapting tests with feedback").

Claim 15:

Johnson and APA discloses the method of claim 17, but do not explicitly disclose wherein a portion of the one or more test cases that corresponds to the one or more desired verification levels does not produce any testing output.

However, APA discloses the stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for example, paragraph [0009]). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Johnson's test case for simple stress tests without producing any output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], "the system does not analyze the output"). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph [0009], "..., so it would be better not to produce it in the first place.")

Claim 16:

Johnson further discloses the method of claim 17, wherein the portion of the one or more test cases that corresponds to the one or more desired verification levels produces one or more test outputs for verifying the software (see for example, Figure 2, step 3-5 of "Project Testing 36", "Record Results", "Report Issues", "Provide Test Case Feedback when necessary" and related text).

Claim 25

Claim 25 is a computer program product version of claimed method in claim 17 above, wherein all claimed limitations have been address and/or set forth above by Johnson and APA. Therefore, as the references teach all the limitation, they also teach the limitations of claim 25. Thus, it also would have been obvious.

Claim 39:

Johnson, Prabhakaran, APA and Bourne's disclose the method of claim 25, but does not discloses wherein the portion of the one or more test cases that corresponds to the one or more desired verification levels does not produce any testing output.

However, APA discloses the stress test that simply ignores any testing output if system doesn't crash when the insert record object is run (see for example, paragraph [0009]). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify and run Johnson's test case for simple stress tests without producing any output. Because, the test output is not important and the system does not analyze the output as pointed out by APA (see for example, page 4, paragraph [0009], "the system does not analyze the output"). One would have been motivated to do so to make test procedure more efficient as suggest by APA (see for example, paragraph [0009], "..., so it would be better not to produce it in the first place.")

Claim 44:

Johnson, Prabhakaran, APA and Bourne's disclose the method as recited in claim 17 Johnson further discloses wherein the method further includes upon detecting and adverse or unexpected result from testing the software, determining of which of the test cases has caused the adverse or unexpected result is accomplished by isolating the plurality of test case within the test group and running each of the isolated test cases individually (see for example, p.2-3, paragraph [0020], "the number of tests and results for tests performed under a predetermined test case or configuration is traceable to view how many times the test case or configuration was used, passed or failed"; also see paragraph [0023], "As tests are run and results recorded, reports are issued to test engineering for tracking test progress and adapting test with feedback"; also see paragraph [0023], "After repetitions of the test cases, test engineers may view results to update test case where testing failures are encouraged by test case faults...").

8. Claims 2-12 and 23-24, 18-20, 26-38, 40 and 42-44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (Johnson et al., US 2004/0073890 A1) in view of Prabhakaran (US 6,859,758) in further view of Bourne (Kelley C. Bourne, Testing Client/Server Systems) and the admitted prior art (APA) of paragraph [0007] of Applicant's background and in further view of Ruffolo (Ruffolo et al., US 2003/0196190 A1).

Claim 2:

Johnson , Prabhakaran, APA and Bourne disclose the method of claim 17, wherein a first test case from the one or more test cases is part of a first test group, the first test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification settings (configurations) that define one or more desired verification levels (Test Iteration) for the first test group (Test Plan) (see for example, Figure 1B, element 30, "Configurations", element 28, "Test Plan", "Test Case", element 26 "Test Iteration" and related text).

But do not disclose the verification settings defining a desired verification level for the one or more test cases. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of "Project Engineering 34" – "Customize Test Cases for the project").

Claim 3:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 2, Johnson further discloses the method comprising acts of:

- identifying a portion of the one or more software testing instructions within the first test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], "A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, "Test Iteration", element 28, "Test Plan", element 30, "Configurations" and related text)
- running a portion of the first test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 "Project Testing" and related detailed steps and text).

Claim 4:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, Johnson also discloses, wherein the verification settings (configurations) define a single desired verification level for the first test case and the first test group (see for example, Figure 1B, "Configuration B" of element 30 "Configurations", using single configuration to cover all test cases in "Test Plan 28", also see related text descriptions).

Art Unit: 2192

Claims 5 and 7:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, but do not explicitly disclose that the verification settings defined verification level for the first/second test cases are different from a desired verification level for the first test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first/second test cases and test group are different, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test case and test group can be different.

Claim 6:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 4, but do not explicitly disclose that the verification settings defined verification level for the second test case are different from a desired verification level for the first test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first/second test cases and test group are different, because each test groups comprises one or more test cases, each test cases can be customized to

different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test case and test group can be different.

Claim 8:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 7, but do not explicitly disclose that the verification settings defined verification level for the first/second test cases is different.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first/second test cases could be different. Because each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases can be different.

Claim 9:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 3, Johnson further discloses wherein a second test case from the one or more test cases is part of the first test group, and wherein third and fourth test cases from the one or more test cases are part of a second test group, the second test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification

settings that define the one or more desired verification levels for the one or more test cases also define one or more desired verification levels for the second test group, the method further comprising acts of:

- identifying a portion of the one or more software testing instructions within the second test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], "A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, "Test Iteration", element 28, "Test Plan", element 30, "Configurations" and related text); and
- running a portion of the second test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 "Project Testing" and related detailed steps and text).

Claim 10:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 9, but do not explicitly disclose that the verification settings defined verification level for the first/second/third/fourth test cases, the first test group and second test group are different.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the test cases and test groups can be set to different verification levels, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test groups can be different.

Claim 11:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 10, Johnson further discloses wherein the first and second test groups are part of a third test group, the third test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, and wherein the verification settings that define the one or more desired verification levels for the one or more test cases also define one or more desired verification levels for the third test group, the method further comprising acts of:

- identifying a portion of the one or more software testing instructions within the second test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], "A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system

having the identified configuration, also see Figure 1B, element 26, "Test Iteration", element 28, "Test Plan", element 30, "Configurations" and related text); and

- running a portion of the second test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 "Project Testing" and related detailed steps and text).

Claim 12:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 9, but do not explicitly disclose that the verification settings define a desired verification level for the third test group different from each of the first test case, the second test case, the third test case, the fourth test case, the first test group and the second test group.

However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the test cases and test groups can be set to different verification levels, because each test groups comprises one or more test cases, each test cases can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test groups can be different.

Art Unit: 2192

Claims 23-24:

Claims 23-24 are a computer program product version of claimed method, wherein all claimed limitations have been address and/or set forth above in claims 2-17. Therefore, as the references teach all the limitation of claims 2-17, they also teach the limitations of claims 23-24 respectively. Thus, they also would have been obvious.

Claim 18:

Johnson, Prabhakaran, APA and Bourne discloses the method of claim 17, wherein a first test case from the one or more test cases is part of a first or a second test group, the first test group including one or more software testing instructions organized as one or more verification levels within the verification hierarchy, further comprising acts of:

- identifying a portion of the one or more software testing instructions within the first test group that corresponds to the one or more desired verification levels (see for example, p.1, paragraph [0010], "A test iteration engine aligns a test case or set of test cases with a configuration to present a matrix view of one or more test cells that guide testing of an information handling system having the identified configuration, also see Figure 1B, element 26, "Test Iteration", element 28, "Test Plan", element 30, "Configurations" and related text); and

- running a portion of the first test group that corresponds to the one or more desired verification levels (see for example, Figure 2, step 36 “Project Testing” and related detailed steps and text)

But does not disclose the verification settings defining a desired verification level for the one or more test cases. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of “Project Engineering 34” – “Customize Test Cases for the project”).

Johnson and Ruffolo also do not explicitly disclose the verification level for the first test case is different form a desired verification level for the first test group. However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first test case and first test group could be different. Because each test groups can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test group can be different.

Claim 19:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 18, wherein a second test case from the one or more test cases is part of the first test group, but do not explicitly disclose the verification level for the second test case is different from a desired verification level for the first test group. However, it would have been obvious to one having ordinary skill in the art at the time the invention was made to understand that the verification levels of the first test case and first test group could be different. Because each test groups can be customized to different verification level to test different degree or portion of software component based on different configurations as discussed above. Therefore, verification levels of the test cases and test group can be different.

Claim 20:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose the method of claim 19, Johnson further discloses wherein verification setting instructions for the desired verification levels define a single verification level for the first and second test cases (see for example, Figure 1B, "Configuration B" of element 30 "Configurations", using single configuration to cover all test cases in "Test Plan 28", also see related text descriptions).

Claims 26-38 and 40:

Claims 26- 38 and 40 are a computer program product version of claimed method in claims 17-20 and 25 above, wherein all claimed limitations have been address and/or set forth above by Johnson and Ruffolo. Therefore, as the references teach all the limitation, they also teach the limitations of claims 25-38 and 40 respectively. Thus, they also would have been obvious.

Claims 42:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose a [The] method as recited in claim 17, but do not explicitly disclose wherein selection of a third verification level causes verification of the record being inserted as well as verification that the record was only inserted a single time and wherein testing of the software includes running the third verification level. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case including just testing a single time insertion. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of "Project Engineering 34" – "Customize Test Cases for the project").

Claim 43:

Johnson, Prabhakaran, APA, Bourne and Ruffolo disclose a [The] method as recited in claim 17, but do not explicitly disclose wherein selection of a third verification level causes verification of the record being inserted as well as verification that the record was inserted without overwriting another record and wherein testing of the software including running the third verification level. However, Ruffolo in the same analogous art of test case generation discloses building different verification level (test items) of test case based on verification settings (distribution list) (see for example, Fig.4, step S406-S412 and relate text). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to define the verification settings for the test case in the configuration file to further customize the verification level of each test case including just testing a single time insertion. One would have been motivated to do so to customize each test case for the project as suggested by Johnson (see for example, Figure 2, step 2a of "Project Engineering 34" – "Customize Test Cases for the project").

Conclusion

11. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.
12. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Zheng Wei whose telephone number is (571) 270-1059 and Fax number is (571) 270-2059. The examiner can normally be reached on Monday-Thursday 8:00-15:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571- 272-1000.

Art Unit: 2192

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Z. W./

Examiner, Art Unit 2192

/Tuan Q. Dam/

Supervisory Patent Examiner, Art Unit 2192